

CLASSIFYING AND TRACKING MULTIPLE PERSONS FOR PROACTIVE SURVEILLANCE OF MASS TRANSPORT SYSTEMS

Suyu Kong¹, C. Sanderson² and Brian C. Lovell^{1,2}

¹ University of Queensland, Brisbane QLD 4072, Australia

² NICTA, 300 Adelaide Street, Brisbane QLD 4000, Australia

Abstract

We describe a pedestrian classification and tracking system that is able to track and label multiple people in an outdoor environment such as a railway station. The features selected for appearance modelling are circular colour histograms for the hue and conventional colour histograms for the saturation and value components. We combine blob matching with a particle filter for tracking and augment these algorithms with colour appearance models to track multiple people in the presence of occlusion. In the object classification stage, hierarchical chamfer matching combined with particle filtering is applied to classify commuters in the railway station into several classes. Classes of interest include normal commuters, commuters with backpacks, commuters with suitcases, and mothers with their children.

1 Introduction

The work presented in this paper is part of a funded project to evaluate research-stage as well as commercially available intelligent surveillance systems. It is part of a larger project to apply intelligent CCTV to enhance counter-terrorism capability for the protection of public spaces and mass transport systems such as rail systems. The aim is to be proactive by responding to incidents before or as they occur, rather than haphazardly reactive as is mostly the case for current security systems.

The system proposed in this paper has two main objectives:

1. To automatically track people in the presence of occlusions.
2. To classify people into classes such as people with suitcases.

Recent developments in the literature include a visual surveillance system called VidMAP [12], that is able to track people and recognize human activities including interactions among people or between people and the environment. Fuentes and Velastin [2] presented a tracking algorithm based on blob matching to track people in surveillance scenarios. However, the approach is limited as it is not able to track multiple persons individually when they

merge. Zhou and Aggarwal [15] also applied blob matching for tracking. Blob size was used to detect occlusions; people tracking during occlusions was accomplished via linear prediction by assuming that the velocity in the current frame was the same as the previous one. This assumption may not be right in some situations, e.g. when two friends meet, they may stop walking and talk with each other. When the blob of the single person splits due to errors in the background segmentation or when several persons merge, blob matching is not robust enough to successfully continue tracking.

There are also many tracking approaches based on particle filtering, e.g. the pioneering work by Isard and Blake [5]. Examples of extensions and related approaches include incorporation of colour information [8], edge information [13], 3D position [9] and 3D human shape models [14]. Perez et al. [10] fused colour information with either sound or motion for teleconferencing and surveillance.

Although particle filters are effective in tracking objects in cases of occlusion or clutter [8], they also have shortcomings. One of them is the initialisation of particles, which needs to be successfully performed before tracking can occur. Initialisation must be automated, as in surveillance systems it is impractical to hand-label a specific person in advance. The reader is directed to the survey written by Hu et al. [4] for more detailed information on recent popular techniques for surveillance and human activity recognition.

In view of the shortcomings of using blob matching or particle filter independently for tracking, we propose an approach that combines these two methods to track multiple people. In our proposed method, the system can recognize persons even when they are partially occluded and keep track of them after merging. In addition, we initialise particle filtering by information obtained in the blob matching stage. Furthermore, a circular statistics method similar to the work by Seitner and Lovell [11] is proposed to analyze the hue component in HSV color space for the background model and the appearance model of humans. Finally, we extend a hierarchical chamfer matching system to detect pedestrians [3] through integration with particle filtering. Fig. 1 shows an overview of the proposed system.

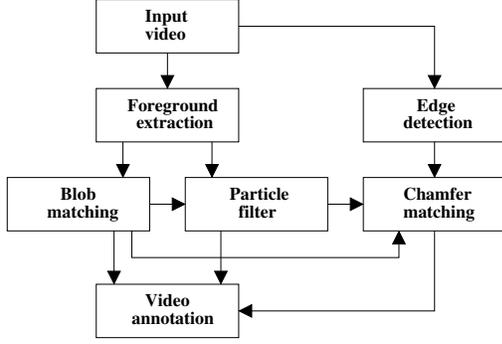


Figure 1. Conceptual block diagram of the proposed tracking and classification system.

The remainder of this paper is organised as follows. In Section 2 we describe the combination of blob matching and particle filtering for robust multiple-person tracking. In Section 3 we describe the integration of particle filtering with hierarchical chamfer matching for commuter classification. Section 4 contains a preliminary evaluation of the proposed approaches on real-life surveillance footage obtained at a railway station. Concluding remarks and avenues for further exploration are given in Section 5.

2 Tracking of people

In this section we introduce a combination of a blob matching algorithm and particle filtering for tracking. The blob matching algorithm is firstly applied to track people. If it fails, particle filtering is applied.

Before commencing people tracking, background subtraction is employed to extract foreground regions. The background subtraction method used in our system is to some extent similar to the work by McKenna et al. [7]. The difference is that while in [7] RGB colour space was used, we use HSV colour space as it is well suited in segmenting chromatic and intensity information. We use a circular statistics method for computing the mean and variance of the hue component, described below.

2.1 Circular statistics

HSV colour space composes of hue, saturation and intensity. Hue is circular in nature while the other two are linear. While it is straightforward to calculate the mean and variance of linear data, computing the mean of the circular data is more involved. For example, two cars move from a common static point. Car A drives off 30° in the north-west direction, where it is assumed that the north direction is 0° . Car A's angle with respect to the north direction will be 330° . Car B drives off 30° in the north-east direction. Using linear statistics, the mean direction will be 180° , which is south, but in actual fact the mean direction

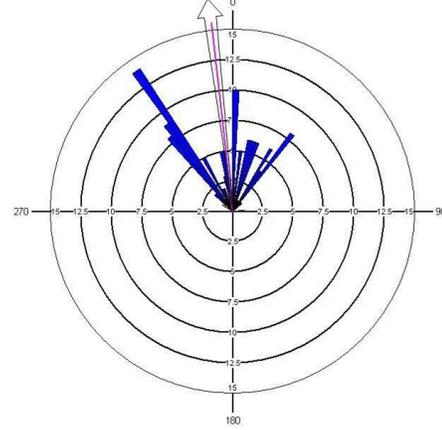


Figure 2. An example of a Rose histogram. The arrow represents the mean value. The length of the blue histogram bar is an indication on the total number of circular data that is within the value range that bar covers, while the number that is located on each circle is an indication on the number of counts on the number of angular data that is within the histogram bar.

should be the north. Fig. 2 shows a set of circular data that is represented as a rose histogram.

According to Mardia [6], the mean and variance of circular data can be calculated as follows. Let $\theta_1, \dots, \theta_{n_1}$ be a random sample of hue values, $\theta_i \in [0, 2\pi)$. Let m_i be the observations of corresponding θ_i , while μ and σ^2 are mean and variance of hue values respectively. The mean can be calculated as:

$$\mu = \left[\arg \left[\sum_{i=1}^{n_1} m_i e^{j\theta_i} \right] \right]_{2\pi} \quad (1)$$

where $[\cdot]_{2\pi}$ denotes reduction modulo 2π onto $[0, 2\pi)$. Based on the von Mises distribution [6], σ^2 can be given as:

$$\sigma^2 = -2 \log_e \left(\frac{1}{n_1} \left| \sum_{i=1}^{n_1} m_i e^{j\theta_i} \right| \right) \quad (2)$$

2.2 Features for tracking

Colour and area information are used as features for the appearance model. In tracking non-rigid objects, colour distribution is invariant to rotation, scale and is robust to partial occlusion [8]. We define the feature vector for the appearance model for Person i as:

$$F_i = \left[H_{i,\{h,s,v\}} \ \mu_{i,\{h,s,v\}} \ A_i \right]' \quad (3)$$

where $H_{i,\{h,s,v\}}$ and $\mu_{i,\{h,s,v\}}$ are the histograms and means of hue, saturation and value respectively, while A_i is the area.

Instead of using a traditional histogram, we apply a method to construct the histogram to allow for small colour aberrations. Below we describe how to build a new histogram for the hue component, since the hue information has a large contribution in the matching score in our system. A traditional histogram $H_i(\theta_i)$ is first constructed using:

$$H_l(\theta_i) = \frac{N(\theta_i)}{\sum_{j=1}^l N(\theta_j)} \quad (4)$$

where $N(\theta_i)$ is the number of pixels whose hue values are equal to θ_i and l is the number of bins of the histogram. A new histogram $H_h(\theta_i)$ is then constructed as follows:

$$H_h(\theta_i) = \frac{\widehat{H}_h(\theta_i)}{\sum_{j=1}^l \widehat{H}_h(\theta_j)} \quad (5)$$

where $\widehat{H}_h(\theta_i) = \sum_{q=-r}^r H_l([\theta_i + q]_l)$ and in turn r is a user defined threshold that is proportional to l .

2.3 Dissimilarity measures

Based on the χ^2 statistic [1], we define the difference $D_H(i, j)$ between the two histograms H_i and H_j as:

$$D_H(i, j) = \sum_{b=1}^l \frac{(H_i(b) - H_m(b))^2}{H_m(b)} \quad (6)$$

where $H_m(b) = (H_i(b) + H_j(b)) / 2$. The difference between the area of two objects is found with:

$$D_A(i, j) = 1 - \left(\frac{\min(A_i, A_j)}{\max(A_i, A_j)} \right)^2 \quad (7)$$

From Eqn. (7), if A_i and A_j are similar, their difference will be close to 0, otherwise, it will be close to 1. Similar to $D_A(i, j)$, the difference between the means of two objects is defined as:

$$D_\mu(i, j) = 1 - \left(\frac{\min(\mu_i, \mu_j)}{\max(\mu_i, \mu_j)} \right)^2 \quad (8)$$

We define the total difference $D(i, j)$ between object i and object j as:

$$D(i, j) = \omega_1 D_H(i, j) + \omega_2 D_A(i, j) + \omega_3 D_\mu(i, j) \quad (9)$$

where ω_1 , ω_2 and ω_3 are corresponding weights. To emphasize the difference between the histograms, we set ω_1 to be larger than ω_2 and ω_3 .

2.4 Blob matching

A two-way matching algorithm is used in our current system. According to $D(i, j)$, firstly, match blobs from the current image with persons detected in the previous images and then match the detected persons with blobs in the current image.

Assuming $B(i)$ represents blob i in the current image while $M(j)$ represents Person j in a set of detected persons. $D_B^M(i, j)$ is the difference between $B(i)$ and $M(j)$, and $D_M^B(i, j)$ is the difference between $M(i)$ and $B(j)$.

Assuming that there are n_3 blobs detected in the current image and n_2 persons in a set of detected people, if $\min(D_B^M(i, 1), D_B^M(i, 2), \dots, D_B^M(i, n_2))$ and $\min(D_M^B(j, 1), D_M^B(j, 2), \dots, D_M^B(j, n_3))$ are equal to $D_B^M(i, j)$ and $D_M^B(j, i)$ respectively, and both values are smaller than a dissimilarity threshold T_1 , then Blob i is assigned to Person j .

Lastly, it should be noted that updating the appearance model is important for continuous tracking. Let G_j be features of Person j detected in the current image. Features for the appearance model of Person j , F_j , are updated with:

$$F_j = (1 - \beta)F_j + \beta G_j \quad (10)$$

where β is the learning rate defining how quickly the old features are forgotten.

2.5 Particle filtering

When the blob matching is finished, the people list is checked to find whether there are people who are not tracked in the current frame. If people are not tracked, there may be some cases. For example, people are occluded by each other, people disappear from the scene, the blob of person is merged with shadow or the blob of person splits into several parts due to bad motion segmentation. Then the particle filter is used to track people not tracked by blob matching, offsetting the shortcoming of the blob matching. Unlike the methods described in Refs. [7, 15], this strategy does not need to check whether people form a group or are under occlusion, but still be capable of tracking people in these difficult situations. In the particle filter, the initialisation of particles for a certain person is based on the information obtained from previous frames in which this person is tracked by either blob matching or particle filtering.

According to [5], when observations accord with a Markov chain, the propagation of state density over time is given as:

$$p(X_t|Z_t) = p(Z_t|X_t)p(X_t|Z_{t-1}) \quad (11)$$

$$p(X_t|Z_{t-1}) = \int_{X_{t-1}} p(X_t|X_{t-1})p(X_{t-1}|Z_{t-1}) \quad (12)$$

where X_t and Z_t is the state of target and the observation of target at time t , respectively.

A particle filter approximates the posterior density $p(X_t|Z_t)$ at time t by a sample set of n particles $S_t = \{(s_t^i, \pi_t^i)\}$, $i = 1, 2, \dots, n$. Each particle s_t^i in state space is associated with a weight π_t^i . A human body is represented as a rectangle, and the vector of state space S_t is defined as $S_t = \{x, y, L_x, L_y, b\}$ where the first two elements represent the centre of the rectangle, L_x and L_y are the width and height of the rectangle, and b is the scale information.

The dynamical model $p(X_t|X_{t-1})$ employed in our system is similar to that in [5]:

$$\widehat{X}_t - \bar{X} = Q(\widehat{X}_{t-1} - \bar{X}) + W_t \quad (13)$$

where $\widehat{X}_t = [X_{t-1} \ X_t]'$, W_t is a Gaussian random variable, \bar{X} is the mean value of the state and Q is the deterministic component of the model. The observation model, $p(Z_t|X_t = s_t^i)$, is decided by the difference $D(i, j)$ between sample i and Person j . Let $D_t(j)$ be the smallest value of $D(i, j)$ among particles. If $D_t(j)$ is smaller than a dissimilarity threshold T_2 , Person j is tracked in the current frame. Features of the appearance model of Person j are then updated using:



Figure 3. Sequential extracts from a surveillance video, with superimposed tracking and labels. In (a) two people merge, while in (b) and (c) three people merge. In (d) the system continues to track and label persons after splitting.

$$F_j = \begin{cases} F_j & \text{if } T_1 < D_t(j) < T_2 \\ (1 - \beta)F_j + \beta G_j & \text{if } D_t(j) \leq T_1 \end{cases}$$

When $D_t(j)$ is between T_1 and T_2 , Person j may be severely occluded. In that case, the features detected in the current image for Person j may be inaccurate and therefore we do not update features of their appearance model.

2.6 Detecting new persons and removing old persons

After two applications of blob matching and particle filtering, pixels in the foreground regions that are already assigned to certain people are then removed. The remaining foreground regions that satisfy certain requirements are considered to be new person candidates. If a candidate is tracked in a predefined number of continuous frames, the candidate is regarded as a confirmed new person. This procedure will make sure that the system tracks a human rather than other moving blobs coming from illumination changes, shadows or reflections, and also will make the initialisation for particles in the step of particle filter reliable. Conversely, old persons are subjects that do not appear in the environment any more. If a person is not tracked in a predefined number of continuous frames, the person is considered as old and then is removed from further consideration.

3 Classification of people

Similar to the work by Gavrilu and Giebel [3], templates for matching are divided into several hierarchy levels in order to accelerate matching. In the matching procedure, we

tracking performance	particles	time
bad	10	0.2344s
reasonable	50	1.0469s
good	100	2.1095s

Table 1. The trade-off between the number of particles and tracking performance. The time indicated is the required time (in Matlab) to track a person whose size is 70×220 pixels. Bad, reasonable and good tracking performance corresponds with loss of tracking of 77%, 22% and 8% during occlusion, respectively.

firstly construct a background model for edges, eliminating the bad effect from edges in the background and therefore making the matching more accurate. A method that is similar to the background subtraction is applied to identify the edges from moving people. Thereafter we convert the current edge image to the distance image by chamfer distance transformation.

The matching process is to some extent similar to [3], but instead of searching the whole image, we propose to employ the particle filter algorithm to search the best position for the template. This has the potential to considerably reduce computational requirements. Compared to the particle filter used for person tracking, the weight of each particle is changed to:

$$\pi_t^i = p(Z_t | X_t = s_t^i) = \frac{\sum_{k=1}^m C_d(k)}{m} \quad (14)$$

where m is total number of edge points in the template and $C_d(k)$ is the value of the point in the distance image, which corresponds to the k -th point in the template. The best position is the one with the smallest weight.

4 Preliminary Evaluations

We performed preliminary evaluations of the proposed system using surveillance footage recently obtained at a railway station in Brisbane, Australia. Firstly, in order to evaluate the performance of the proposed tracking approach, it was tested on video sequences where there are several people merging together. A subset of the results is shown in Fig. 3, where it can be seen that people are correctly labelled and tracked before and during merging; the tracking continues correctly after splitting.

When particle filtering is employed to track people during occlusion, processing time increases greatly compared to blob matching. We note that fast execution can be an important aspect, especially when there is a multitude of people to track. We can reduce the number of particles in order to reduce the computational demands, however this comes at the expense of less accurate tracking results. See Tab. 1 for an example.

We have also tested our chamfer matching method for commuter classification. By building a background model for edges, edges related to the background are removed before chamfer distance transformation, making our method



Figure 4. Commuter classification results. (a): results from the method described in [3]. (b): results from the proposed technique.



Figure 5. Examples of the detection of a commuter with luggage.

robust to background clutter. Fig. 4 shows a subset of the results obtained by the proposed technique as well as the method described in [3]. In [3], there is no stationary background and as such there are errors by chamfer matching due to the background clutter. Additional methods are required to verify the matching results. In the proposed system, bad effects from background clutter are removed.

When applying particle filtering to search the best position for the template, we assign 70 particles to each person, which means it only searches 70 positions and requires $\sum_{i=1}^{70} R_i$ correlations between template and image to classify each person, where R_i is the number of templates needed to be matched in position i based on the hierarchical architecture of templates in the silhouette dataset. In our videos, the size of each frame is 704×576 .

The number of particles can be decreased to reduce the processing time. However, similarly to the previously mentioned trade-off, this comes at the expense of worse matching. An example is given in Tab. 2. Fig. 5 shows resulting images where a commuter with luggage is detected successfully by searching no more than 70×4 positions in the image.

matching performance	particles	time
bad	5	0.0052s
reasonable	30	0.0313s
good	70	0.0938s

Table 2. The trade-off between the number of particles and matching performance. Indicated time is the time spent (in Matlab) to match a person with one template whose size is 109×200 pixels. Bad, reasonable and good matching performance corresponds to accuracies of 25%, 70% and 89%, respectively.

5 Concluding Remarks

A combination of a blob matching algorithm and particle filtering for tracking and labelling multiple people in the presence of occlusion has been proposed. Circular statistics is applied to analyse the hue component in the HSV colour space. In addition, a commuter classification approach based on hierarchical chamfer matching integrated with particle filtering was introduced. Preliminary results indicate that the proposed system is robust in tracking multiple people under occlusion and can accurately classify commuters. Apart from a more thorough evaluation, future work also includes adding more silhouette templates into our dataset and applying a semantic codebook for labelling.

Acknowledgements

This project is supported by a grant from the Australian Government Department of the Prime Minister and Cabinet. NICTA is funded by the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council.

References

- [1] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2001.
- [2] L. Fuentes and S. Velastin. People tracking in surveillance applications. *Image and Vision Computing*, 24(11):1165–1171, 2006.
- [3] D. Gavrilu and J. Giebel. Shape-based pedestrian detection and tracking. In *Proc. Intelligent Vehicle Symposium*, volume 1, pages 8 – 14, 2002.
- [4] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. Systems, Man and Cybernetics, Part C*, 34(3):334–352, 2004.
- [5] M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.
- [6] K. Mardia. *Statistics of directional data*. Academic Press, London, 1972.
- [7] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler. Tracking groups of people. *Computer Vision and Image Understanding*, 80(1):42–56, 2000.
- [8] K. Nummiaro, E. Koller-Meier, and L. Van Gool. An adaptive color-based particle filter. *Image and Vision Computing*, 21(1):99–110, 2003.
- [9] T. Osawa, X. Wu, K. Wakabayashi, and T. Yasuno. Human tracking by particle filtering using full 3d model of both target and environment. In *ICPR*, pages 25–28, 2006.
- [10] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proc. of the IEEE*, 92(3):495 – 513, 2004.
- [11] F. Seitner and B. Lovell. Pedestrian tracking based on colour and spatial information. In *DICTA*, pages 36 – 43, 2005.
- [12] V. Shet, D. Harwood, and L. Davis. VidMAP: video monitoring of activity with Prolog. In *AVSS*, pages 224–229, 2005.
- [13] C. Yang, R. Duraiswami, and L. Davis. Fast multiple object tracking via a hierarchical particle filter. In *ICCV*, pages 212–219, 2005.
- [14] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. In *CVPR*, pages 406–413, 2004.
- [15] Q. Zhou and J. Aggarwal. Object tracking in an outdoor environment using fusion of features and cameras. *Image and Vision Computing*, 24(11):1244–1255, 2006.